全国青少年信息学奥林匹克竞赛

CCF NOI 2025

第一试

时间：2025 年 7 月 14 日 08:00 ∼ 13:00

题目名称	机器人	序列变换	数字树
题目类型	传统型	传统型	传统型
目录	`robot`	`sequence`	`tree`
可执行文件名	`robot`	`sequence`	`tree`
输入文件名	`robot.in`	`sequence.in`	`tree.in`
输出文件名	`robot.out`	`sequence.out`	`tree.out`
每个测试点时限	1.0 秒	6.0 秒	4.0 秒
内存限制	1024 MiB	1024 MiB	1024 MiB
测试点数目	20	20	25
测试点是否等分	是	是	是
预测试点数目	20	20	25

提交源程序文件名

对于 C++ 语言	`robot.cpp`	`sequence.cpp`	`tree.cpp`

编译选项

对于 C++ 语言	`-O2 -std=c++14 -static`

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。赛后正式测试时将以选手留在题目目录下的源代码为准。
2. `main` 函数的返回值类型必须是 `int`，程序正常结束时的返回值必须是 0。
3. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末换行）。
4. 选手提交的程序源文件大小不得超过 100 KiB。
5. 程序可使用的栈空间内存限制与题目的内存限制一致。
6. 禁止在源代码中改变编译器参数（如使用 `#pragma` 命令），禁止使用系统结构相关指令（如内联汇编）或其他可能造成不公平的方法。
7. 因违反上述规定而出现的问题，申诉时一律不予受理。
8. 选手可使用快捷启动页面中的工具 selfEval 进行自测。选手需将待测程序的源文件置于相应题目目录下。每次自测时可选择全部或部分题目进行自测。注意：自测有次数限制，且自测结果仅供选手测试参考，不作为最终正式成绩。

机器人（robot）

【题目描述】

NOI2025 正在绍兴举办，小 Y 为闭幕式表演制作了一个机器人并打算操控它从仓库走到礼堂。

绍兴的道路系统可以简化为 n 个路口以及连接这些路口的 m 条**单行道路**，且每条道路有一定的长度。为了方便将道路系统录入机器人的芯片，小 Y 对每一个路口连接的所有道路进行了编号。具体而言，若有 d 条道路以路口 x 为起点，则这 d 条道路会被小 Y 按照某种顺序编号为 $1 \sim d$，分别称作以 x 为起点的第 $1 \sim d$ 条道路。

小 Y 的机器人内部有一个参数 p。给定参数 p 的上限 k 与修改费用 $v_1, v_2, \ldots, v_{k-1}$, $w_2, w_3, \ldots, w_k$。小 Y 将按照如下规则设置与修改机器人的参数：

- 初始时，小 Y 将参数 p 设置为 1。
- 在**任意时刻**，小 Y 可以远程控制机器人修改参数：
  - 若 $p < k$，则小 Y 可以花费 v_p 的费用将 p 增加 1，即 $p \leftarrow p+1$；
  - 若 $p > 1$，则小 Y 可以花费 w_p 的费用将 p 减少 1，即 $p \leftarrow p-1$。

初始时，小 Y 的机器人位于机器人仓库，即路口 1。当机器人位于路口 x 时，记以路口 x 为起点的第 p 条道路的终点为 y，道路长度为 z，则小 Y 可以花费 z 的费用操控机器人从 x 走到 y。特别地，若以路口 x 为起点的道路不足 p 条，则小 Y 无法操控机器人走动。

小 Y 并不知道闭幕式表演所在的礼堂位于哪个路口，因此他需要对每个路口都做好准备。请你帮助他求出将机器人从仓库移动到每个路口所需费用的最小值。

【输入格式】

从文件 ***robot.in*** 中读入数据。

输入的第一行包含一个非负整数 c，表示测试点编号。$c = 0$ 表示该测试点为样例。

输入的第二行包含三个正整数 n, m, k，分别表示路口数量、道路数量与参数 p 的上限。

输入的第三行包含 $k-1$ 个非负整数 $v_1, \ldots, v_{k-1}$，表示增加参数 p 的费用。

输入的第四行包含 $k-1$ 个非负整数 $w_2, \ldots, w_k$，表示减少参数 p 的费用。

输入的第 $i+4$ $(1 \le i \le n)$ 行包含若干个正整数，其中第一个非负整数 d_i 表示以路口 i 为起点的道路数量，接下来 $2d_i$ 个正整数 $y_{i,1}, z_{i,1}, y_{i,2}, z_{i,2}, \ldots, y_{i,d_i}, z_{i,d_i}$，表示以路口 i 为起点的道路，其中 $y_{i,j}, z_{i,j}$ $(1 \le j \le d_i)$ 分别表示编号为 j 的道路的终点与长度。

【输出格式】

输出到文件 ***robot.out*** 中。

输出一行 n 个整数，其中第 i $(1 \le i \le n)$ 个数表示小 Y 将机器人从仓库移动到路口 i 所需费用的最小值。特别地，若小 Y 无法将机器人从仓库移动到该路口，则输出 -1。

【样例 1 输入】

```
0
5 6 3
2 4
1 1
3 2 5 3 1 4 2
1 3 2
2 1 2 4 1
0
0
```

【样例 1 输出】

```
0 5 3 4 -1
```

【样例 1 解释】

小 Y 可以按照以下方案将机器人分别从仓库移动到路口 $1 \sim 4$：

- 对于路口 1：小 Y 的机器人初始时即位于路口 1，因此所需费用为 0；
- 对于路口 2：小 Y 操控机器人沿以路口 1 为起点的第 1 条道路走到路口 2，所需费用为 5；
- 对于路口 3：小 Y 将参数 p 增加 1，然后操控机器人沿以路口 1 为起点的第 2 条道路走到路口 3，所需费用为 $2 + 1 = 3$；
- 对于路口 4：小 Y 将参数 p 增加 1，然后操控机器人沿以路口 1 为起点的第 2 条道路走到路口 3，再操控机器人沿以路口 3 为起点的第 2 条道路走到路口 4，所需费用为 $2 + 1 + 1 = 4$。

可以证明，上述移动方案的所需费用均为最小值。

- 对于路口 5：由于小 Y 无法将机器人移动到路口 5，因此输出 -1。

【样例 2】

见选手目录下的 ***robot/robot2.in*** 与 ***robot/robot2.ans***。

该样例满足测试点 $3 \sim 5$ 的约束条件。

【样例 3】

见选手目录下的 ***robot/robot3.in*** 与 ***robot/robot3.ans***。
该样例满足测试点 $6 \sim 8$ 的约束条件。

【样例 4】

见选手目录下的 ***robot/robot4.in*** 与 ***robot/robot4.ans***。
该样例满足测试点 $9, 10$ 的约束条件。

【样例 5】

见选手目录下的 ***robot/robot5.in*** 与 ***robot/robot5.ans***。
该样例满足测试点 $16 \sim 18$ 的约束条件。

【数据范围】

对于所有测试数据，保证：

- $1 \le n, m \le 3 \times 10^5$，$1 \le k \le 2.5 \times 10^5$；
- 对于所有 $1 \le i \le k-1$，均有 $0 \le v_i \le 10^9$；
- 对于所有 $2 \le i \le k$，均有 $0 \le w_i \le 10^9$；
- 对于所有 $1 \le i \le n$，均有 $0 \le d_i \le k$，且 $\sum_{i=1}^{n} d_i = m$；
- 对于所有 $1 \le i \le n$，$1 \le j \le d_i$，均有 $1 \le y_{i,j} \le n$，$1 \le z_{i,j} \le 10^9$。

<table>
<tr><th>测试点编号</th><th>$n, m \le$</th><th>$k \le$</th><th>特殊性质</th></tr>
<tr><td>$1, 2$</td><td>6</td><td>6</td><td rowspan="2">C</td></tr>
<tr><td>$3 \sim 5$</td><td>10^3</td><td>10^3</td></tr>
<tr><td>$6 \sim 8$</td><td>5×10^4</td><td>10^2</td><td>无</td></tr>
<tr><td>$9, 10$</td><td rowspan="4">10^5</td><td rowspan="4">10^5</td><td>AB</td></tr>
<tr><td>$11, 12$</td><td>A</td></tr>
<tr><td>$13 \sim 15$</td><td>C</td></tr>
<tr><td>$16 \sim 18$</td><td rowspan="2">无</td></tr>
<tr><td>$19, 20$</td><td>3×10^5</td><td>2.5×10^5</td></tr>
</table>

特殊性质 A：保证 $v_1 = v_2 = \cdots = v_{k-1} = 0$ 且 $w_2 = w_3 = \cdots = w_k = 0$。
特殊性质 B：保证对于所有 $1 \le i \le n$，$1 \le j \le d_i$，均有 $z_{i,j} = 1$。
特殊性质 C：保证至多存在 10 个 i 满足 $d_i \ge 10$。

序列变换（sequence）

【题目描述】

给定两个长度为 n 的整数序列 $B=[b_1,\ldots,b_n]$，$C=[c_1,\ldots,c_n]$。对于长度为 n 的非负整数序列 $D=[d_1,\ldots,d_n]$，设 $S(D)$ 为所有满足 $d_i=0$ 的下标 i 的集合，定义 $f(D)=\sum_{i\in S(D)} b_i$，$g(D)=\prod_{i\in S(D)} c_i$。特别地，若 $S(D)$ 为空，则 $f(D)=0$，$g(D)=1$。

小 L 有一个长度为 n 的**正整数序列** $A=[a_1,\ldots,a_n]$。小 L 可以对序列 A 做如下修改：

- 选择序列 A 的两个**相邻**的下标 i,j（即 $1\le i,j\le n$ 且 $|i-j|=1$），若 $a_i\le a_j$，则将 a_j 改为 a_j-a_i，同时将 a_i 改为 0。

小 L 可以进行任意多次修改操作，也可以不进行任何修改。对于所有序列 A 通过以上修改操作可以得到的序列 D，小 L 想求出 $f(D)$ 的最大值以及 $g(D)$ 之和，请你帮助他求出这两个值。形式化地，记 $T(A)$ 为序列 A 通过以上修改操作可以得到的**所有序列的集合**，你需要求出 $\max_{D\in T(A)} f(D)$ 以及 $\sum_{D\in T(A)} g(D)$。其中，由于 $\sum_{D\in T(A)} g(D)$ 可能较大，你只需要求出其对 $1,000,000,007$ 取模后的结果。

【输入格式】

从文件 ***sequence.in*** 中读入数据。

本题包含多组测试数据。

输入的第一行包含两个非负整数 c,t，分别表示测试点编号与测试数据组数。$c=0$ 表示该测试点为样例。

接下来依次输入每组测试数据，对于每组测试数据：

第一行包含一个正整数 n，表示序列长度。

第二行包含 n 个正整数 $a_1,\ldots,a_n$，表示序列 A。

第三行包含 n 个整数 $b_1,\ldots,b_n$，表示序列 B。

第四行包含 n 个正整数 $c_1,\ldots,c_n$，表示序列 C。

【输出格式】

输出到文件 ***sequence.out*** 中。

对于每组测试数据，仅输出一行，其中包含两个整数，分别表示 $\max_{D\in T(A)} f(D)$ 以及 $\sum_{D\in T(A)} g(D)$ 对 $1,000,000,007$ 取模后的结果。**注意**：$\max_{D\in T(A)} f(D)$ **不需要对** $1,000,000,007$ **取模**。

本题包含两个小问，正确回答其中任意一个小问均可获得部分分数。具体评分规则请参见【评分方式】。

【样例 1 输入】

```
0 3
3
5 6 6
3 6 9
1 2 3
6
1 1 4 5 1 4
-1 1 -1 1 -2 2
1 1 1 1 1 1
8
4 2 4 2 2 2 4 4
-2 4 9 -3 4 8 7 8
1 1 1 1 1 1 1 1
```

【样例 1 输出】

```
15 10
1 18
37 48
```

【样例 1 解释】

该样例共包含三组测试数据。

对于第一组测试数据，可以得到以下 4 个序列：

- $D=[5,6,6]$，$f(D)=0$，$g(D)=1$；
- $D=[0,1,6]$，$f(D)=3$，$g(D)=1$；
- $D=[5,0,0]$，$f(D)=6+9=15$，$g(D)=2\times 3=6$；
- $D=[0,0,5]$，$f(D)=3+6=9$，$g(D)=1\times 2=2$。

故 $\max_{D\in T(A)} f(D)=\max\{0,3,15,9\}=15$，$\sum_{D\in T(A)} g(D)=1+1+6+2=10$。

【样例 2】

见选手目录下的 *sequence/sequence2.in* 与 *sequence/sequence2.ans*。

该样例满足测试点 3, 4 的约束条件。

【样例 3】

见选手目录下的 ***sequence/sequence3.in*** 与 ***sequence/sequence3.ans***。
该样例满足测试点 $5, 6$ 的约束条件。

【样例 4】

见选手目录下的 ***sequence/sequence4.in*** 与 ***sequence/sequence4.ans***。
该样例满足测试点 7 的约束条件。

【样例 5】

见选手目录下的 ***sequence/sequence5.in*** 与 ***sequence/sequence5.ans***。
该样例满足测试点 $11, 12$ 的约束条件。

【样例 6】

见选手目录下的 ***sequence/sequence6.in*** 与 ***sequence/sequence6.ans***。
该样例满足测试点 $16 \sim 18$ 的约束条件。

【数据范围】

设 N 为单个测试点内所有测试数据的 n 的和。对于所有测试数据，保证：

- $1 \le t \le 20$；
- $1 \le n \le 5,000$，$N \le 4 \times 10^4$；
- 对于所有 $1 \le i \le n$，均有 $1 \le A_i \le 10^9$；
- 对于所有 $1 \le i \le n$，均有 $-10^9 \le B_i \le 10^9$；
- 对于所有 $1 \le i \le n$，均有 $1 \le C_i \le 10^9$。

<table>
<tr><th>测试点编号</th><th>$n \le$</th><th>$N \le$</th><th>特殊性质</th></tr>
<tr><td>$1, 2$</td><td>8</td><td>10^2</td><td>无</td></tr>
<tr><td>$3, 4$</td><td rowspan="2">200</td><td rowspan="2">400</td><td>B</td></tr>
<tr><td>$5, 6$</td><td>无</td></tr>
<tr><td>7</td><td rowspan="3">500</td><td rowspan="3">10^3</td><td>A</td></tr>
<tr><td>$8 \sim 10$</td><td>B</td></tr>
<tr><td>$11, 12$</td><td>无</td></tr>
<tr><td>13</td><td rowspan="3">$3,500$</td><td rowspan="3">3×10^4</td><td>A</td></tr>
<tr><td>$14, 15$</td><td>B</td></tr>
<tr><td>$16 \sim 18$</td><td rowspan="2">无</td></tr>
<tr><td>$19, 20$</td><td>$5,000$</td><td>4×10^4</td></tr>
</table>

特殊性质 A：保证 $A_1 = A_2 = \cdots = A_n = 1$。

特殊性质 B：保证对于所有 $1 \le i \le n$，A_i 均在 $[1, 10^9]$ 中**独立均匀随机**生成。

【评分方式】

对于每个测试点：

- 正确回答所有测试数据的 $\max_{D \in T(A)} f(D)$，可获得该测试点 40% 的分数；
- 正确回答所有测试数据的 $\sum_{D \in T(A)} g(D)$ 对 $1,000,000,007$ 取模后的结果，可获得该测试点 60% 的分数。

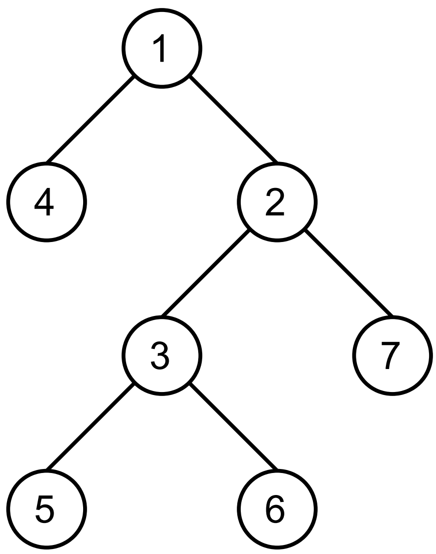
注意：即使选手仅回答了其中一个问题，也需要按照输出格式输出两个整数，分别对应两个问题的答案。

数字树（tree）

【题目描述】

给定一棵 $4n-1$ 个结点的二叉树，其中每个非叶结点都有恰好两个子结点。非叶结点编号为 1 到 $2n-1$，叶子结点编号为 $2n$ 到 $4n-1$。初始时，每个叶子结点上都没有数字。

定义一个 DFS 序是优美的，当且仅当按该 DFS 序将所有标有数字的叶子结点上的数字拼成一个序列时，该序列可以通过若干次消除相邻相同数字的方式得到空序列。例如，在下图中，若叶子结点 $4,6$ 上标有数字 1，叶子结点 $5,7$ 上标有数字 2，则按 DFS 序 $[1,4,2,7,3,5,6]$ 将所有标有数字的叶子结点上的数字拼成的序列为 $[1,2,2,1]$，可以通过消除相邻的 2 的方式得到 $[1,1]$，再通过消除相邻的 1 的方式得到空序列，因此该 DFS 序是优美的；而按 DFS 序 $[1,4,2,3,5,6,7]$ 将所有标有数字的叶子结点上的数字拼成的序列为 $[1,2,1,2]$，无法通过若干次消除相邻相同数字的方式得到空序列，因此该 DFS 序不是优美的。



给定 n 次操作，第 i $(1 \le i \le n)$ 次操作会选择两个没有数字的叶子结点，然后将这两个结点标上数字 i。**保证在每次操作后，存在至少一个优美的 DFS 序。**你需要求出每次操作后的优美的 DFS 序的数量。由于答案可能较大，你只需要求出答案对 $1,000,000,007$ 取模后的结果。

【输入格式】

从文件 ***tree.in*** 中读入数据。

输入的第一行包含一个非负整数 c，表示测试点编号。$c=0$ 表示该测试点为样例。

输入的第二行包含一个正整数 n，表示二叉树的结点个数为 $4n-1$。

输入的第 $i+2$ $(1 \le i \le 2n-1)$ 行包含两个正整数 l_i 和 r_i，分别表示结点 i 的左右子结点。保证 $i < l_i, r_i \le 4n-1$，且所有的 l_i, r_i 互不相同。

输入的第 $i+2n+1$ $(1 \le i \le n)$ 行包含两个正整数 a_i, b_i，表示第 i 次操作选择的叶子结点的编号。保证 $2n \le a_i, b_i \le 4n-1$，且所有的 a_i, b_i 互不相同。

【输出格式】

输出到文件 ***tree.out*** 中。

输出 n 行，其中第 i $(1 \le i \le n)$ 行包含一个非负整数，表示第 i 次操作后的优美的 DFS 序的数量对 $1,000,000,007$ 取模后的结果。

【样例 1 输入】

```
0
2
4 2
3 7
5 6
4 6
5 7
```

【样例 1 输出】

```
8
4
```

【样例 1 解释】

该样例即【题目描述】中所示的例子。

- 第一次操作后，叶子结点 4 和 6 上标有数字 1，叶子结点 5 和 7 上没有数字，因此按任意 DFS 序拼成的序列均为 $[1,1]$，即所有的 $2^3 = 8$ 个 DFS 序都是优美的。
- 第二次操作后，叶子结点 $4 \sim 7$ 上分别标有数字 $1,2,1,2$，因此共有 4 个优美的 DFS 序，分别为 $[1,4,2,3,6,5,7]$，$[1,4,2,7,3,5,6]$，$[1,2,3,6,5,7,4]$，$[1,2,7,3,5,6,4]$。

【样例 2 输入】

```
0
6
```

```
2 3
4 21
22 23
5 11
6 8
7 9
12 13
10 18
14 15
16 17
19 20
12 13
14 15
16 19
17 18
20 21
22 23
```

【样例 2 输出】

```
2048
2048
2048
1024
512
512
```

【样例 3】

见选手目录下的 *tree/tree3.in* 与 *tree/tree3.ans*。
该样例满足测试点 $6 \sim 10$ 的约束条件。

【样例 4】

见选手目录下的 *tree/tree4.in* 与 *tree/tree4.ans*。
该样例满足测试点 $11, 12$ 的约束条件。

【样例 5】

见选手目录下的 ***tree/tree5.in*** 与 ***tree/tree5.ans***。
该样例满足测试点 $17 \sim 20$ 的约束条件。

【样例 6】

见选手目录下的 ***tree/tree6.in*** 与 ***tree/tree6.ans***。
该样例满足测试点 $24, 25$ 的约束条件。

【数据范围】

对于所有测试数据，保证：

- $1 \le n \le 2 \times 10^5$；
- 对于所有 $1 \le i \le 2n-1$，均有 $i < l_i, r_i \le 4n-1$，且所有的 l_i, r_i 互不相同；
- 对于所有 $1 \le i \le n$，均有 $2n \le a_i, b_i \le 4n-1$，且所有的 a_i, b_i 互不相同；
- 在每次操作后，存在至少一个优美的 DFS 序。

测试点编号	$n \le$	特殊性质
$1, 2$	10	无
$3 \sim 5$	10^2	A
$6 \sim 10$		无
$11, 12$	10^3	A
$13, 14$		无
$15, 16$	5×10^4	AB
$17 \sim 20$		B
$21, 22$		无
23	2×10^5	A
$24, 25$		无

特殊性质 A：保证每次操作选择的两个叶子结点位于结点 1 的不同子树内。

特殊性质 B：保证存在非负整数 m 满足 $n = 2^m$，且对于所有 $1 \le i \le 2n-1$，均有 $l_i = 2i$，$r_i = 2i+1$。